WHAT IS CLAIMED IS:

1. A controller for managing a plurality of response packets and a plurality of request packets between interconnection rings and at least one memory access controller, the controller includes a table with a plurality of entries, each entry maintains a register of information about a particular packet from the plurality of response packets and the plurality of request packets, the table comprises:

means for storing a table entry state which defines a current status of a entry of the plurality of entries;

means for storing a flow type which defines a transaction being performed by the controller with the particular packet;

means for storing a master identification which designates an originator of the packet; and

means for storing a transaction identification which designates the packet for the originator;

wherein the information stored in the table is used by the controller to direct its operations.

2. The controller of claim 1 wherein:

the master identification and the transaction identification together form a unique identifier for each packet.

3. The controller of claim 1 wherein:

the table entry state is selected from the group consisting of unused, queued, waiting, queue waiting, active, and done.

4. The controller of claim 1 wherein:

the flow type is selected from the group consisting of read shared, read private, rollout, dflush, read current, write purge, global flush, increment update, and non-coherent.

5. The controller of claim 4 further comprising:

means for storing rollout phase data for resolving collisions between a flow having the rollout flow type and another flow having the increment update flow type.

6. The controller of claim 1 further comprising:

means for queuing request information until interconnection rings are accessible; and

means for queuing response information until the at least one memory access controller is accessible.

7. The controller of claim 6 further comprising:

means for constructing request packets from the request information.

8. The controller of claim 6 further comprising:

means for constructing response packets from the response information.

9. The controller of claim 6 wherein said controller implements a Scalable Coherent Interface (SCI) protocol.

10. A method of managing, by a controller, a plurality of entries that are each associated with a respective packet of a plurality of response packets and a plurality of requests packets communicated between an interconnection ring of a multi-node processor system and at least one memory access controller, the method comprising:

storing entry states that define current status information for each respective packet;

storing flow types that define a transaction being performed by the controller for each respective packet;

storing master identification information that designates an originator of each respective packet; and

storing transaction identification information that identifies each respective packet for its originator;

wherein said controller uses said entry states, flow types, master identification information, and transaction identification information to communicate packets between said at least one memory controller and said interconnection ring.

11. The method of claim 10 wherein a combination of a master identification and a transaction identification forms a unique identifier for a respective packet.

12. The method of claim 10 wherein said entry states are selected from the group consisting of: unused, queued, waiting, queue waiting, active, and done.

13. The method of claim 10 wherein the flow types are selected from the group consisting of: read shared, read private, rollout, dflush, read current, write purge, global flush, increment update, and non-coherent.

14. The method of claim 13 further comprising:

storing rollout phase data for resolving collisions between a flow having a rollout flow type and another flow having an increment update flow type.

15. The method of claim 10 further comprising:

managing a request activation queue that stores request information until said interconnection ring is accessible; and

managing a response activation queue that stores response information until the at least one memory access controller is accessible.

16. The method of claim 15 further comprising:

assembling request packets from said request information.

17. The controller of claim 15, further comprising:

assembling response packets from the response information.

18. A controller for managing a plurality of response packets and a plurality of request packets between interconnection ring of a multi-node processor system and at least one memory access controller, the controller comprising:

a table structure for storing, for each respective packet, an entry state that defines current status information, a flow type that define a transaction being performed by the controller, master identification information that identifies an originator, transaction identification information defined by a respective originator;

a request activation queue for storing information until said interconnection ring is accessible; and

a response activation queue for storing response information until the at least one memory access controller is accessible.

19. The controller of claim 18 further comprising:

a request packet assembler for constructing request packets from the request information.

20. The controller of claim 18 further comprising:

a response engine for constructing response data packets from the response information.